

Zadatak 1.

Opisati jezik koji se generiše pomoću sledeće gramatike sa startnim simbolom $\langle S \rangle$.

1. $\langle S \rangle \rightarrow 10\langle S \rangle 0$
2. $\langle S \rangle \rightarrow a\langle A \rangle$
3. $\langle A \rangle \rightarrow b\langle A \rangle$
4. $\langle A \rangle \rightarrow a$

Rešenje

Razmotrimo najpre skup sentenci koje se mogu izvesti iz neterminala $\langle A \rangle$. Višestrukim primenama 3., pa zatim 4. smene dobija se:

$$\langle A \rangle \Rightarrow^* b^m a, \quad m \geq 0$$

Višestrukim primenama 1. smene, pa zatim primenom 2. smene dobija se:

$$\langle S \rangle \Rightarrow^* (10)^n a \langle A \rangle 0^n, \quad n \geq 0$$

Kada u obzir uzmemo sentence koje je moguće generisati iz $\langle A \rangle$, dobijamo konačno rešenje:

$$L(G) = \{(10)^n a b^m a 0^n \mid m, n \geq 0\}$$

Zadatak 2.

Pronaći bezkontekstnu gramatiku koja generiše sledeće jezike:

- a) $\{1^n 0^m\} \quad n > m > 0$
- b) $\{1^n 0^n 1^m 0^m\} \quad m, n \geq 0$
- c) $\{1^n 0^m 1^m 0^n\} \quad m, n \geq 0$
- d) $\{1^{3n+2} 0^n\} \quad n \geq 0$

Rešenje

- a) Najkraća sentenca u skupu $\{1^n 0^m\}$, $n > m > 0$ je 110. Duže sentence mogu se dobiti repetitivnim dodavanjem jedne jedinice na početak sentence, ili istovremeno jedne jedinice na početak i jedne nule na kraj:

1. $\langle S \rangle \rightarrow 110$
2. $\langle S \rangle \rightarrow 1\langle S \rangle 0$
3. $\langle S \rangle \rightarrow 1\langle S \rangle$

- b) Sentence oblika $1^n 0^n 1^m 0^m$, $m, n \geq 0$ sastoje se iz dva podniza istog oblika $1^k 0^k$, $k \geq 0$ koji se opisuju posebno uvedenim neterminalom $\langle S_1 \rangle$:

1. $\langle S \rangle \rightarrow \langle S_1 \rangle \langle S_1 \rangle$
2. $\langle S_1 \rangle \rightarrow 1\langle S_1 \rangle 0$
3. $\langle S_1 \rangle \rightarrow \epsilon$

- c) Posebnim neterminalom $\langle S_1 \rangle$ opisujemo podnizove oblika $0^m 1^m$, $m \geq 0$. Sentence dobijamo repetitivnim dodavanjem jedinice na početak i nule na kraj, što je definisano pravilima za startni neterminal $\langle S \rangle$:

1. $\langle S \rangle \rightarrow 1\langle S \rangle 0$
2. $\langle S \rangle \rightarrow \langle S_1 \rangle$
3. $\langle S_1 \rangle \rightarrow 0\langle S_1 \rangle 1$
4. $\langle S_1 \rangle \rightarrow \epsilon$

- d) Tražena gramatika glasi:

1. $\langle S \rangle \rightarrow 11$
2. $\langle S \rangle \rightarrow 111\langle S \rangle 0$

Zadatak 3.

Napisati gramatiku koja opisuje niz od jednog ili više brojeva. Prikazati stablo izvođenja za ulaz 3 7 12.

Rešenje

Da bi se opisao niz uzastopnih neterminala potrebno je koristiti rekurziju (levu ili desnu). Gramatika je:

$$\langle \text{niz} \rangle \rightarrow \langle \text{niz} \rangle \text{NUMBER}$$

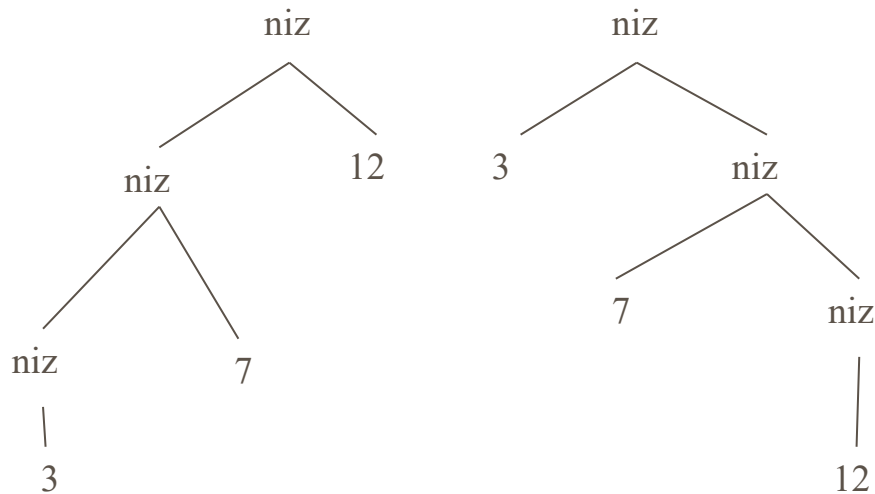
$$\langle \text{niz} \rangle \rightarrow \text{NUMBER}$$

ili

$$\langle \text{niz} \rangle \rightarrow \text{NUMBER} \langle \text{niz} \rangle$$

$$\langle \text{niz} \rangle \rightarrow \text{NUMBER}$$

Stabla izvođenja za levu i desnu rekurziju su:



Napomena:

Ove smene se mogu koirstiti za opisivanje tela funkcija, kao skupa iskaza:

`<body> → <body> <statement>`

`<body> → <statement>`

Zadatak 4.

Napisati gramatiku koja opisuje niz od jednog ili više članova koji mogu biti brojevi, identifikatori ili datumi.

Rešenje

Da bi se opisao niz uzastopnih neterminala potrebno je koristiti rekurziju (levu ili desnu) . Pošto član koji se ponavlja može biti raznih tipova, u posebnoj smeni se definiše šta može da bude član. Gramatika je:

`<list> → <list> <item>`

`<list> → <item>`

`<item> → NUMBER`

`<item> → IDENTIFIKATOR`

`<item> → DATUM`

Napomena:

Ove smene se mogu koirstiti za opisivanje elemanata glavnog programa, kao skupa deklaracija promenljivih, konstanti i funkcija:

`<program> → <program> <program_item>`

`<program> → <program_item>`

`<program_item> → <var_decl>`

`<program_item> → <const_decl>`

`<program_item> → <funct_decl>`

Ovakvu formu ima i telo funkcije kao niz iskaza koji mogu biti iskazi dodele, pozivi funkcija, while petlje itd...

`<body> → <body> <statement>`

`<body> → <statement>`

`<statement> → <assignment>`

`<statement> → <procedure_call>`

`<statement> → <while_loop>`

Zadatak 5.

Napisati gramatiku koja opisuje niz od jednog ili više brojeva međusobno odvojenih zarezima. Prikazati stablo izvođenja za ulaz 3,7,12.

Rešenje

Da bi se opisao niz uzastopnih neterminala potrebno je koristiti rekurziju (levu ili desnu) . Pošto član koji se ponavlja može biti raznih tipova, u posebnoj smeni se definiše šta može da bude član. Gramatika je:

`<csv> → <csv> , NUMBER`

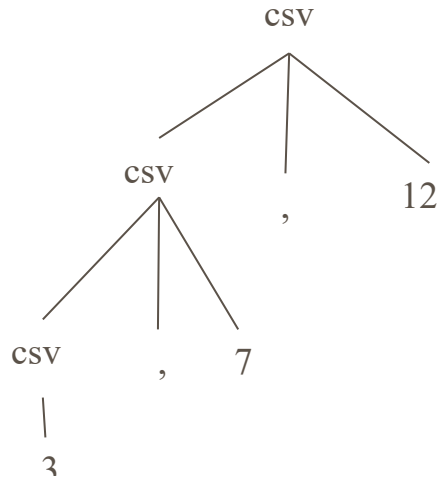
`<csv> → NUMBER`

ili

`<csv> → NUMBER, <csv>`

`<csv> → NUMBER`

Stablo izvođenja za prvu gramatiku(sa levom rekurzijom)za ulaz 3,7,12 je:



Napomena:

Ove smene se mogu koristiti za opisivanje niza deklaracija formalnih parametara u deklaraciji funkcije. Primer za deklaracije C funkcija oblika *int saberi(int x, int y, int z);* je

$\langle f_decl \rangle \rightarrow \langle type \rangle \text{ IDENTIFIKATOR } (\langle arg_list \rangle) ;$

$\langle arg_list \rangle \rightarrow \langle arg_list \rangle , \langle arg_def \rangle$

$\langle arg_list \rangle \rightarrow \langle arg_def \rangle$

$\langle arg_def \rangle \rightarrow \langle type \rangle \text{ IDENTIFIKATOR}$

Ove smene se mogu koristiti i u deklaracijama promenljivih (oblika *int x,y,z;*). Gramatika za ovaj slučaj je:

$\langle var_decl \rangle \rightarrow \langle type \rangle \langle var_list \rangle ;$

$\langle var_list \rangle \rightarrow \langle var_list \rangle , \text{ IDENTIFIKATOR}$

$\langle var_list \rangle \rightarrow \text{ IDENTIFIKATOR}$

Ove smene se mogu koristiti i u pozivima funkcija (oblika *proc(x,y,z);*). Gramatika za ovaj slučaj je:

$\langle proc_call \rangle \rightarrow \text{ IDENTIFIKATOR } (\langle var_list \rangle);$

$\langle var_list \rangle \rightarrow \langle var_list \rangle , \langle var \rangle$

$\langle var_list \rangle \rightarrow \langle var \rangle$

Ove smene se mogu koristiti i u referenciranju polja objekata (oblika *this.array.length*). Gramatika za ovaj slučaj je:

$\langle designator \rangle \rightarrow \langle designator \rangle . \text{ IDENTIFIKATOR}$

$\langle designator \rangle \rightarrow \text{ IDENTIFIKATOR}$

Zadatak 6.

Napisati gramatiku koja opisuje aritmetičke izraze koji sadrže samo operacije sabiranja pri čemu je sabiranje levo asocijativna operacija. Napisati gramatiku koja podržava aritmetičke izraze u slučaju da je operacija sabiranja desno asocijativna.

Rešenje

Aritmetički izrazi predstavljaju niz sabiraka međusobno odvojenih znakom +. Gramatika je kombinacija gramatika iz zadataka 4 i 5. Pošto je rečeno da je sabiranje levo asocijativna operacija, **mora** se koristiti leva rekurzija:

$\langle zbir \rangle \rightarrow \langle zbir \rangle + \langle sabirak \rangle$

$\langle zbir \rangle \rightarrow \langle sabirak \rangle$

$\langle sabirak \rangle \rightarrow \text{ NUMBER}$

$\langle sabirak \rangle \rightarrow \text{ IDENTIFIKATOR}$

U slučaju da je sabiranje desno asocijativno koristila bi se desna rekurzija:

$\langle zbir \rangle \rightarrow \langle sabirak \rangle + \langle zbir \rangle$

$\langle zbir \rangle \rightarrow \langle sabirak \rangle$

$\langle sabirak \rangle \rightarrow \text{ NUMBER}$

$\langle sabirak \rangle \rightarrow \text{ IDENTIFIKATOR}$

Zadatak 7.

Napisati gramatiku koja opisuje aritmetičke izraze koji sadrže operacije sabiranja i oduzimanja pri čemu su sabiranje i oduzimanje **levo** asocijativne operacije istog prioriteta.

Rešenje

Aritmetički izrazi predstavljaju niz sabiraka međusobno odvojenih znacima + i -. Pošto je rečeno da su sabiranje i oduzimanje **levo** asocijativne operacije, **mora** se koristiti leva rekurzija:

$\langle zbir \rangle \rightarrow \langle zbir \rangle + \langle sabirak \rangle$

$\langle zbir \rangle \rightarrow \langle zbir \rangle - \langle sabirak \rangle$

$\langle zbir \rangle \rightarrow \langle sabirak \rangle$

$\langle sabirak \rangle \rightarrow \text{ NUMBER } | \text{ IDENTIFIKATOR}$

Zadatak 8.

Napisati gramatiku koja opisuje aritmetičke izraze koji sadrže operacije sabiranja i množenja pri čemu su sabiranje i množenje **levo** asocijativne operacije, i množenje ima veći prioritet nad sabiranjem. Prikazati stablo izvođenja za ulaz $2*x+3$.

Rešenje

Aritmetički izrazi predstavljaju niz sabiraka međusobno odvojenih znakom $+$. Gramatika je kombinacija gramatika iz zadataka 4 i 5. Pošto je rečeno da su sabiranje i oduzimanje **levo** asocijativne operacije, **mora** se koristiti leva rekurzija:

$\langle \text{zbir} \rangle \rightarrow \langle \text{zbir} \rangle + \langle \text{proizvod} \rangle$

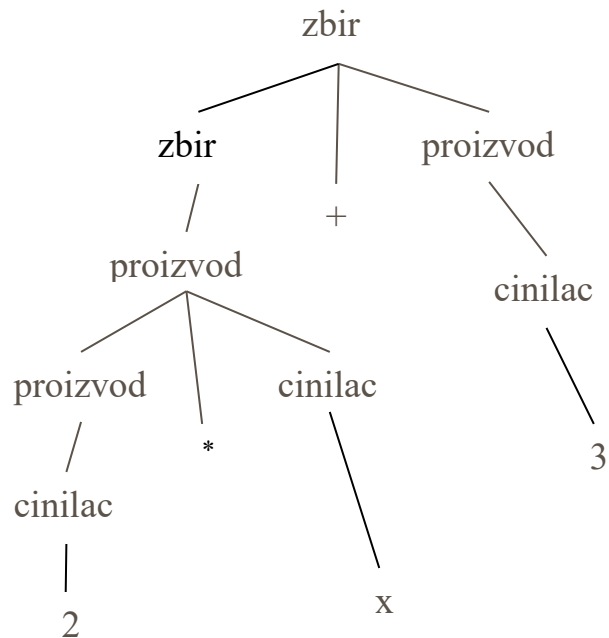
$\langle \text{zbir} \rangle \rightarrow \langle \text{proizvod} \rangle$

$\langle \text{proizvod} \rangle \rightarrow \langle \text{proizvod} \rangle * \langle \text{cinilac} \rangle$

$\langle \text{proizvod} \rangle \rightarrow \langle \text{cinilac} \rangle$

$\langle \text{cinilac} \rangle \rightarrow \text{NUMBER} \mid \text{IDENTIFIKATOR}$

Stablo izvođenja za ulaz $2*x+3$:



Napomena:

Ove smene se mogu koristiti za opisivanje logičkih izraza. Primer za deklaracije C funkcija oblika *int saberi(int x, int y, int z);* je

$\langle \text{disjunction} \rangle \rightarrow \langle \text{disjunction} \rangle \text{ OR } \langle \text{conjunction} \rangle$

$\langle \text{disjunction} \rangle \rightarrow \langle \text{conjunction} \rangle$

$\langle \text{conjunction} \rangle \rightarrow \langle \text{conjunction} \rangle \text{ AND } \langle \text{condition} \rangle$

$\langle \text{conjunction} \rangle \rightarrow \langle \text{condition} \rangle$

$\langle \text{condition} \rangle \rightarrow \langle \text{var} \rangle \text{ RELOP } \langle \text{var} \rangle$